

2- Notas Históricas

Linguagem	Ano	Originador	LPs Predecessoras	Características
Plankalkül	1945	Zuse	-	<ul style="list-style-type: none">• Cálculos de Programas
FORTRAN	1954- 1957	Backus	-	<ul style="list-style-type: none">• Introduziu expressões simbólicas e subprogramas com parâmetros• Computações numéricas
ALGOL60	1958- 1960	ANSI	FORTRAN	<ul style="list-style-type: none">• Introduziu o conceito de bloco• Computações numéricas

COBOL	1959-1960	ANSI	-	<ul style="list-style-type: none"> • Introduziu o conceito de descrição de dados • Aplicações comerciais
LISP	1956-1962	McCarthy	-	<ul style="list-style-type: none"> • Computação simbólica baseada em funções sobre listas e árvores
SNOBOL	1962	Griswold	-	<ul style="list-style-type: none"> • Processamento de sequências
PL/I	1963	Comitê da IBM	FORTTRAN ALGOL60 COBOL	<ul style="list-style-type: none"> • Tentativa de projetar uma LP genérica
SIMULA67	1967	Dahl	ALGOL60	<ul style="list-style-type: none"> • Simulação de maneira geral
ALGOL68	1968	ANSI	ALGOL60	<ul style="list-style-type: none"> • Permite a combinação sistemática de um conjunto de conceitos • Programação de sistemas
PASCAL	1971	Wirth	ALGOL60	<ul style="list-style-type: none"> • LP mais popular da família ALGOL • Simples, sistemática e de implementação eficiente • Propósito geral e de ensino

PROLOG	(±) 1970	Colmerauer	-	<ul style="list-style-type: none"> • Popularização da programação em lógica • Fraco e ineficiente em sua forma pura
C	1974	Ritchie	ALGOL68	<ul style="list-style-type: none"> • Programação de sistemas
CONCURRENT PASCAL	1975	Brinch	PASCAL	<ul style="list-style-type: none"> • Programação concorrente
MODULA	1977	Wirth	PASCAL	<ul style="list-style-type: none"> • Programação de sistemas em tempo real

ADA	1979	Ichbiah et al.	SIMULA67 PASCAL	<ul style="list-style-type: none"> • Introduziu pacotes, funções genéricas, tratamento de alto nível para exceções e concorrência • Propósito geral
SMALLTALK80	1983	Goldberg e Robson	SIMULA	<ul style="list-style-type: none"> • Baseada em classes de objetos
C++	(±) 1983	Stroustrup	C SIMULA67	<ul style="list-style-type: none"> • Programação orientada a objetos
ML	(±) 1985	Harper	LISP	<ul style="list-style-type: none"> • Programação funcional mais moderna

MIRANDA	(±) 1985	Turner	ML	• Programação funcional mais moderna
EIFFEL	1988	Meyer	SIMULA	• Programação orientada a objetos
Outras: DELPHI, JAVA, C#, Python, Haskell, etc	1990's	-	-	• LP's mais modernas
Outras: Rust, Kotlin, Go, JavaScript...	2000's	-	-	• LP's mais modernas

Características presentes na evolução de LP's:

- Maior nível de abstração
- Proliferação de paradigmas de programação

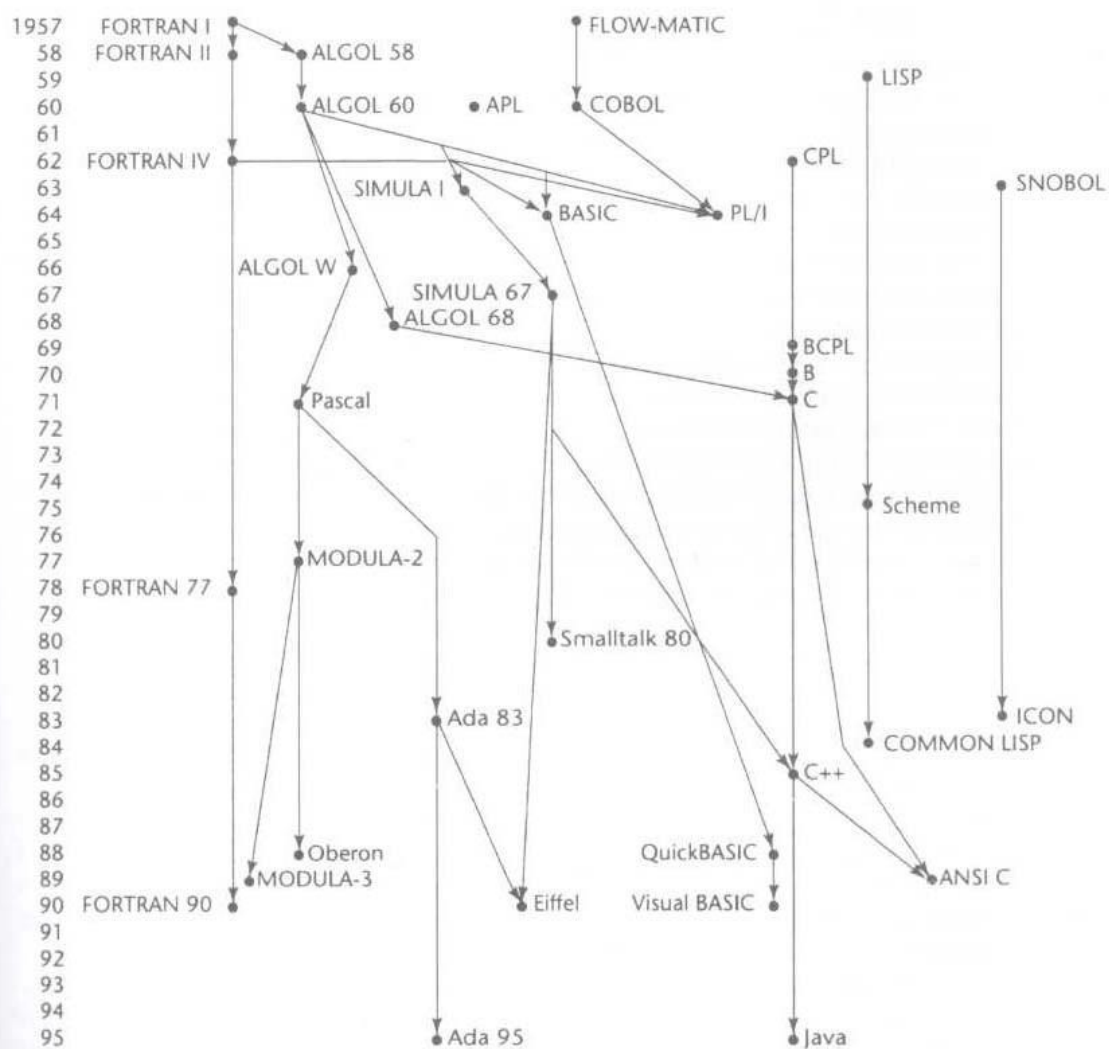


FIGURA 2.1 Genealogia das linguagens de programação de alto nível comuns.